CSE 231—Advanced Operating Systems “Ballooning”

Andrew Quinn

Background. The VMWare ESX paper [?] focuses on overcoming the memory issues in managing memory on a Type-1 hypervisor. The approaches presented in the paper aim seamlessly manage the memory of multiple virtual machines executing on large hardware resources.

The most “famous” approach presented in the paper is the ballooning technique. Traditionally, a type-1 hypervisor will implement paging below each operating system, so that each operating system is unaware of whether each page of memory is located on disk or in-memory. Unfortunately, VMM paging interacts poorly with each OS’s paging; correct VMM paging decisions are counter-intuitive. For example, if the VMM paging system evicts a page under memory pressure and the OS evicts the same page, then the system will incur two extra page faults.

VMWare’s solution is a balloon driver that inflates when VMM faces memory pressure. Specifically, the balloon driver requests physical memory from the operating system (they use the term “pinning” memory in the paper) and therefore takes some of the physical memory that was originally assigned to the guest. When memory pressure decreases, the balloon deflates by relinquishing its physical memory.

Discussion. I have five discussion questions/comments about this paper; the last is based upon discussion questions from Surya.

1. **What are the security impacts of content based sharing?** A VM can use a timing side-channel to determine if another VM is using a specific page. With no additional information, this remains a brute-force attack. But, the attacker could learn which code is in use by checking for code text matches. Then, they could make an informed search based upon the common data-types used by the application.

2. **Is Ballooning just a form of paravirtualizaiton?**

3. **This paper "looks" different from the rest.** This paper described a number of smaller techniques and evaluated each one in isolation. In contrast, most papers that we have looked at presented a single grand vision and have a single evaluation section. What do you think about these different stylistic choices? Do you prefer one over the other?

4. **Why is this paper best known through the ballooning technique?** Should it be?

5. **Is the content-based sharing actually effective?** It seems odd in that it seems like it will only be effective in scenarios in which all VMs use the same operating system. Figure 5 shows that their example deployments use the same OSes... Doesn't it seem like a weird use of virtualization, though?